

BEST AVAILABLE COP

P.A. Y

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 10-040158

(43)Date of publication of application : 13.02.1998

(51)Int.Cl.

G06F 12/00

G06F 17/30

(21)Application number : 08-213024

(71)Applicant : FUJI XEROX CO LTD

(22)Date of filing : 23.07.1996

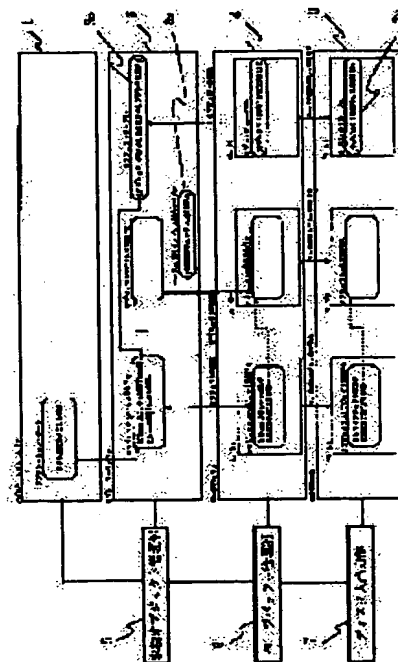
(72)Inventor : HAYATA HIROSHI  
WATANABE MIKI

## (54) PERMANENT OBJECT MANAGING METHOD

## (57)Abstract:

**PROBLEM TO BE SOLVED:** To exhibit advantages that respective identifiers have and to improve the throughput by distinguishably using a position identifier and an object identifier which are set for a permanent object.

**SOLUTION:** In accordance with the maintenance cost needed to maintain reference consistency based upon the attribute of the permanent object, the object identifier or position identifier is set for the permanent object and when a permanent object to which an object has already being identifier loaded on a main storage device 3 is set is referred to, an object table 3o is searched with the object identifier indicating a reference destination to obtain an address. The permanent object on the main storage device 3 is referred to with this address, and when the permanent object to which the position identifier loaded onto the main storage device 3 is set is referred to, an address correspondence table 3a is searched with the position identifier indicating the reference destination to obtain an address, and the permanent object on the main storage device 3 is referred to with the address.



## LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's  
decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

P.A. 8

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-40158

(43) 公開日 平成10年(1998) 2月13日

(51) Int.Cl. <sup>6</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 12/00 17/30	5 4 7		G 0 6 F 12/00 15/40	5 4 7 A 3 8 0 E

審査請求 未請求 請求項の数 4 F D (全 13 頁)

(21) 出願番号 特願平8-213024

(22) 出願日 平成 8 年(1996) 7 月23日

(71) 出願人 000005496

富士ゼロックス株式会社  
東京都港区赤坂二丁目17番22号

(72) 発明者 早田 宏

神奈川県足柄上郡中井町境430 グリーン  
テクなかい 富士ゼロックス株式会社内

(72) 発明者 渡辺 美樹

神奈川県足柄上郡中井町境430 グリーン  
テクなかい 富士ゼロックス株式会社内

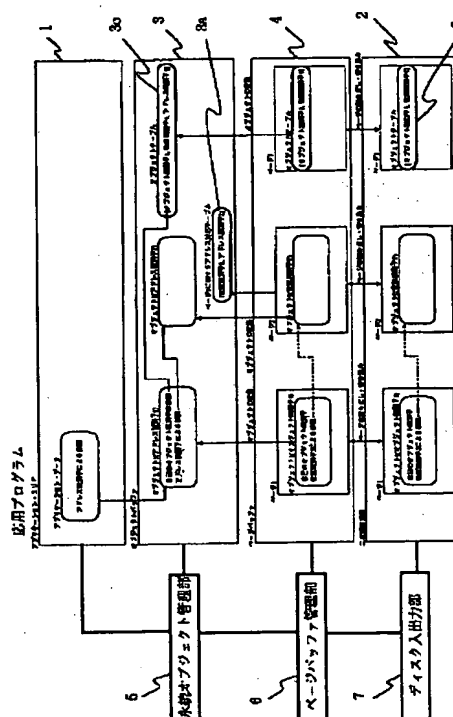
(74) 代理人 弁理士 守山 辰雄

(54) 【発明の名称】 永続オブジェクト管理方法

(57) 【要約】

【課題】 永続オブジェクトに設定する位置識別子とオブジェクト識別子とを使い分けることにより、それぞれの識別子の持つ利点を発揮させて処理効率を向上させる。

【解決手段】 永続オブジェクトの属性に基づく参照一貫性を保つためにかかる保守コストによって、永続オブジェクトにオブジェクト識別子或いは位置識別子を設定し、さらに、既に主記憶装置 3 上にロードされているオブジェクト識別子が設定された永続オブジェクトを参照する際には、参照先を示すオブジェクト識別子によってオブジェクトテーブル 3 o を探索してアドレスを取得し、当該アドレスにより主記憶装置 3 上の永続オブジェクトを参照する一方、既に主記憶装置 3 上にロードされている位置識別子が設定された永続オブジェクトを参照する際には、参照先を示す位置識別子によってアドレス対応テーブル 3 a を探索してアドレスを取得し、当該アドレスにより主記憶装置 3 上の永続オブジェクトを参照する。



## 【特許請求の範囲】

【請求項1】 永続オブジェクトを識別子によって識別して参照するオブジェクト指向データベースにおいて、永続オブジェクトを生成する際に、当該永続オブジェクトの属性に基づく参照一貫性を保つためにかかる保守コストが所定値以上であることに基づいて一意の論理的な値のオブジェクト識別子を設定し、保守コストが所定値未満であることに基づいて当該オブジェクトが格納される二次記憶装置での格納位置を示す位置識別子を設定することを特徴とする永続オブジェクト管理方法。

【請求項2】 請求項1に記載した永続オブジェクト管理方法において、オブジェクト識別子が設定されている永続オブジェクトを主記憶装置上にロードする際には、当該永続オブジェクトがロードされる主記憶装置上のアドレスをオブジェクト識別子に対応付けて主記憶装置上のオブジェクトテーブルに格納する一方、位置識別子が設定されている永続オブジェクトを主記憶装置上にロードする際には、当該永続オブジェクトがロードされる主記憶装置上のアドレスを位置識別子に対応付けて主記憶装置上のアドレス対応テーブルに格納しておき、オブジェクト識別子によって永続オブジェクトを参照する際には、オブジェクトテーブルを探索し、当該オブジェクト識別子に対応するアドレスがある場合には当該アドレスによって主記憶装置上の永続オブジェクトを参照し、位置識別子によって永続オブジェクトを参照する際には、アドレス対応テーブルを探索し、当該位置識別子に対応するアドレスがある場合には当該アドレスによって主記憶装置上の永続オブジェクトを参照することを特徴とする永続オブジェクト管理方法。

【請求項3】 請求項2に記載した永続オブジェクト管理方法において、オブジェクト識別子が設定された永続オブジェクトが主記憶装置上にロードされた場合に、主記憶装置上の当該永続オブジェクトが当該永続オブジェクトのオブジェクト識別子に対応する主記憶装置上のオブジェクトテーブル内のエントリのアドレスを保持することを特徴とする永続オブジェクト管理方法。

【請求項4】 請求項1乃至請求項3のいずれか1項に記載の永続オブジェクト管理方法において、バックアップされた永続オブジェクトを二次記憶装置に再成する場合には、当該永続オブジェクトがバックアップされた際に設定されていたオブジェクト識別子を当該永続オブジェクトに再度設定することを特徴とする永続オブジェクト管理方法。

## 【発明の詳細な説明】

## 【0001】

【発明の属する技術分野】 本発明は、オブジェクト指向データベースにおいて、永続オブジェクトを識別する方

法、永続オブジェクトを参照する方法といった永続オブジェクトの管理方法に関する。

## 【0002】

【従来の技術】 従来よりデータとデータを操作する手続きとを合わせたオブジェクトを管理するオブジェクト指向データベースが考えられており、複雑化、多様化したデータを容易にモデル化できるとともに効率的に管理できるために、文書やマルチメディアデータ等といったデータを管理する分野において特に注目されている。

【0003】 このオブジェクト指向データベースの特徴の1つは、一のオブジェクトが他のオブジェクトを参照していることを示すオブジェクト間の参照関係を個々のオブジェクトに割り付けた識別子によって表現し、当該割り付けた識別子によって一のオブジェクトから他のオブジェクトへ辿ることができることであり、参照関係の表現の仕方としては、例えば参照元のオブジェクトから参照先のオブジェクトへの参照を表す場合には、参照元のオブジェクト内のメンバ変数に参照先のオブジェクトの識別子を保持する等によって表現している。

【0004】 上記した識別子は、ただ単に適当に割り付けられればよいものでなく、オブジェクトを確実に識別して参照できるといった参照一貫性を確保しなければならない。すなわち、違うオブジェクトに対して同一の識別子を割り当てしまう等といったオブジェクトが識別できない状況を避けなければならない。

【0005】 上記したオブジェクトを確実に識別するといった識別性に関しては、Addison-Wesley Publishing社R.G.G.Cattel著「Object Data Management Revised Edition」(1994) (以下、文献1という。)の85～86頁ならびに151～158頁に記載されている。

【0006】 永続オブジェクトに割り当てる識別子の一つとしては、永続オブジェクトの二次記憶装置上の格納位置によって決まる位置識別子を用いる方式があり、この方式では参照元の永続オブジェクトが参照先の永続オブジェクトの位置識別子を保持し、当該位置識別子によって参照先のオブジェクトの二次記憶装置上での位置を即座に特定して参照することができる。

【0007】 また、永続オブジェクトに割り当てる他の識別子として、一意の論理的な値のオブジェクト識別子を用いる方式があり、この方式では図6に示すように永続オブジェクトの二次記憶装置上での位置と永続オブジェクトに割り当てたオブジェクト識別子との組を管理するオブジェクトテーブル11を主記憶装置上(オブジェクトバッファ上)に保持することによって、参照元のオブジェクトが参照先のオブジェクトを参照することができる。すなわち、オブジェクトテーブル11を探索して、参照元の永続オブジェクトが保持している参照先の永続オブジェクトを表すオブジェクト識別子に対応する位置識別子をオブジェクトテーブル11から取得し、当該位置識別子によって、二次記憶装置上の参照先の永続

10

20

30

40

50

オブジェクトを参照することができる。なお、通常、上記したオブジェクトテーブル11は二次記憶装置上で永続化されたオブジェクトテーブル12として格納される。

【0008】例えば、“オブジェクト1”が“オブジェクト識別子2”が割り当てられた“オブジェクト2”を参照する場合においては、“オブジェクト1”は“オブジェクト2”に割り当てられた“オブジェクト識別子2”を保持しており、当該“オブジェクト識別子2”に対応する“位置識別子2”を主記憶装置上のオブジェクトテーブル11から取得し、取得した“位置識別子2”によって二次記憶装置上の“オブジェクト2”を参照することが可能になる。

【0009】また、オブジェクト指向データベースでは、永続オブジェクトを二次記憶装置から主記憶装置にロードして読み出し操作や書き込み操作を行っており、例えば、複数の永続オブジェクトが或る一つの永続オブジェクトを参照する場合や、一つの永続オブジェクトが或る一つの永続オブジェクトを複数回参照する場合に、同一の永続オブジェクトを重複してロードする状況が発生しており、このような重複したロードによって処理効率の低下や、異なるデータを持った永続オブジェクトが複数主記憶装置上に存在してしまうといった問題も生じていた。

【0010】この問題に対して、特開平5-225034号公報に記載された発明では、主記憶装置上に一度ロードされた永続オブジェクトのアドレスを表すアドレス識別子をオブジェクト・テーブルで管理する方法と、主記憶装置上の参照元の永続オブジェクトが参照先の永続オブジェクトのアドレス識別子を保持する方法と、が実施されている。この発明では、図6に示すようにオブジェクトテーブル11として、オブジェクト識別子と位置識別子とアドレス識別子との3つ組を管理し、主記憶装置上に永続オブジェクトがロードされている場合にはアドレス識別子を登録しておき、当該アドレス識別子によって主記憶装置上の永続オブジェクトを参照することによって、重複した永続オブジェクトのロードを回避して参照一貫性を保証するとともに、永続オブジェクトの参照処理の効率を向上している。

【0011】また、この発明では主記憶装置上にロードされた永続オブジェクトの構造（以下、メモリイメージという。）をプログラミング言語によって生成される揮発オブジェクトと、永続オブジェクト管理の情報をもつハンドラとの組で構成し、ハンドラによって主記憶装置上の永続オブジェクトへの参照等の操作を行っている。

【0012】また、J. E. Moss著「Working with Persistent Objects: To Swizzle or Notto Swizzle」IEEE Transaction on Software Engineering Vol.18 No.8 (1992)では、メモリイメージでの参照を表現するためのアドレス識別子に関する変換方式が開示されている。

【0013】

【発明が解決しようとする課題】従来の位置識別子のみを用いたオブジェクト指向データベースにおいては、永続オブジェクトを参照する際には、位置識別子から直ちに二次記憶装置上の参照先の永続オブジェクトを特定して参照することができ、参照の効率がいい。しかしながら、この位置識別子を用いたオブジェクト指向データベースでは、二次記憶装置上の永続オブジェクトの位置を変更する場合或いは永続オブジェクト自体を削除する場合には、変更或いは削除する永続オブジェクトを参照しているすべて永続オブジェクトが保持する位置識別子を変更或いは削除する必要が生じ、手間がかかってしまう。

【0014】これに対して、オブジェクト識別子のみを用いたオブジェクト指向データベースでは、永続オブジェクトの位置の変更する場合や、永続オブジェクト自体の削除する場合には、参照元の永続オブジェクトに保持されている参照先を表すオブジェクト識別子を変更する必要はなく、オブジェクトテーブルに管理されているオブジェクト識別子と位置識別子との対応を変更するだけでよく、手間がかからない。

【0015】しかしながら、オブジェクト識別子のみを用いたオブジェクト指向データベースでは、オブジェクト識別子と当該オブジェクト識別子の割り当てられたオブジェクトの二次記憶装置上の位置とをオブジェクトテーブルとして管理する必要があり、永続オブジェクトの数が多くなるとオブジェクトテーブルが大きくなってしまい、主記憶装置上或いは二次記憶装置の記憶領域が多く必要となってしまう。また、オブジェクトテーブルが大きくなればなるほど、参照の際に行う探索の効率或いは時間といった探索コストが大きくなる。さらに、複数ユーザが同時に永続オブジェクトの参照を実施する場合においては、オブジェクトテーブルでの探索がコンフリクトして処理の並列性が低下し、結果として、処理効率が低下してしまう。

【0016】一般に、オブジェクト指向データベースでは、重複した永続オブジェクトの主記憶装置上へのロードを回避して、永続オブジェクトに対する参照一貫性を保証するとともに、永続オブジェクトの参照処理の効率を向上することが望まれている。

【0017】また、オブジェクト指向データベースでは、永続オブジェクトの保全のために、二次記憶上のデータベースファイルから、特定の永続オブジェクトをテープなどの三次記憶装置にバックアップし、その後、三次記憶装置から二次記憶装置にリストアすることを可能にする必要があり、バックアップ後にリストアした永続オブジェクトへの参照がバックアップ時と同様に維持される必要がある。

【0018】そこで、本発明は上記した従来の事情に鑑みなされたもので、位置識別子を用いて行う永続オブジェクト管理方法と、オブジェクト識別子を用いて行う永

続オブジェクト管理方法とを使い分けることにより、双方の利点を発揮させて処理効率を向上することのできる永続オブジェクト管理方法を提供することを目的としている。

【0019】また、本発明は上記したオブジェクト識別子及び位置識別子を使い分けるオブジェクト指向データベースにおいて、重複した永続オブジェクトの主記憶装置上へのロードを回避して、永続オブジェクトに対する参照一貫性を保つとともに処理効率を向上させることのできる永続オブジェクト管理方法を提供することを目的としている。また、本発明は上記したオブジェクト識別子及び位置識別子を使い分けるオブジェクト指向データベースにおいて、保全性を向上することのできる永続オブジェクト管理方法を提供することを目的としている。

#### 【0020】

【課題を解決するための手段】上記目的を達成するために本発明は、永続オブジェクトの属性に基づいて参照一貫性を保つためにかかる保守コストに着目してなされたもので、保守コストが所定値以上である場合には、永続オブジェクトに対して一意の論理的なオブジェクト識別子を設定して保守コストをできるだけ抑え、保守コストが所定値未満である場合には、永続オブジェクトに対して二次記憶装置上の格納位置を示す位置識別子を設定して永続オブジェクトの参照効率を向上する。

【0021】ここで、永続オブジェクトの属性に基づいて参照一貫性を保つためにかかる保守コストとは、永続オブジェクトの二次記憶装置上の位置を移動する処理、永続オブジェクトを二次記憶装置上から削除する処理等を行う際に、処理の前後において各永続オブジェクト間の参照一貫性を崩さず保つためにかかる手間等といったコストを意味している。このため、例えば、多くの永続オブジェクトから参照される永続オブジェクトは保守コストが高く、一つ或いは少数の永続オブジェクトからしか参照されない永続オブジェクトは保守コストは低い。また、一度作成した後に移動或いは削除されないといった永続オブジェクトの保守コストは低く、頻繁に格納位置を変える可能性のある永続オブジェクトは保守コストが高いことがいえる。なお、所定値は任意で良く、オブジェクト指向データベースのユーザが設定することもできるし、応用プログラムが設定することもできる。

【0022】そこで、本発明の請求項1に係る永続オブジェクト管理方法は、永続オブジェクトを識別子によって識別して参照するオブジェクト指向データベースにおいて、永続オブジェクトを生成する際に、当該永続オブジェクトの属性に基づく参照一貫性を保つためにかかる保守コストが所定値以上であることに基づいて一意の論理的な値のオブジェクト識別子を設定し、保守コストが所定値未満であることに基づいて当該オブジェクトが格納される二次記憶装置での格納位置を示す位置識別子を設定することを特徴とする。

【0023】上記した請求項1に記載した永続オブジェクト管理方法では、永続オブジェクトの属性に基づく参照一貫性を保つためにかかる保守コストによって、永続オブジェクトにオブジェクト識別子か、位置識別子かを設定することができる。したがって、オブジェクト指向データベースデータベースにおいて参照一貫性を保つためにかかる保守コストを削減することができ、また、オブジェクト識別子が設定される永続オブジェクトの数も抑えることができるので、オブジェクトテーブルの大きさを縮小することができ、主記憶装置及び二次記憶装置上の領域を有効に利用することができるとともに、オブジェクト識別子によって永続オブジェクトを参照する際にオブジェクトテーブルでの探索コストを低減することができる。

【0024】本発明の請求項2に係る永続オブジェクト管理方法は、請求項1に記載した永続オブジェクト管理方法において、オブジェクト識別子が設定されている永続オブジェクトを主記憶装置上にロードする際には、当該永続オブジェクトがロードされる主記憶装置上のアドレスをオブジェクト識別子に対応付けて主記憶装置上のオブジェクトテーブルに格納する一方、位置識別子が設定されている永続オブジェクトを主記憶装置上にロードする際には、当該永続オブジェクトがロードされる主記憶装置上のアドレスを位置識別子に対応付けて主記憶装置上のアドレス対応テーブルに格納しておき、オブジェクト識別子によって永続オブジェクトを参照する際には、オブジェクトテーブルを探索し、当該オブジェクト識別子に対応するアドレスがある場合には当該アドレスによって主記憶装置上の永続オブジェクトを参照し、位置識別子によって永続オブジェクトを参照する際には、アドレス対応テーブルを探索し、当該位置識別子に対応するアドレスがある場合には当該アドレスによって主記憶装置上の永続オブジェクトを参照することを特徴とする。

【0025】上記した請求項2に係る永続オブジェクト管理方法では、既に主記憶装置上にロードされているオブジェクト識別子が設定された永続オブジェクトを参照する際には、参照先を示すオブジェクト識別子によってオブジェクトテーブルを探索してアドレスを取得し、当該アドレスにより主記憶装置上の永続オブジェクトを参照する一方、既に主記憶装置上にロードされている位置識別子が設定された永続オブジェクトを参照する際には、参照先を示す位置識別子によってアドレス対応テーブルを探索してアドレスを取得し、当該アドレスにより主記憶装置上の永続オブジェクトを参照する。

【0026】したがって、参照する永続オブジェクトが主記憶装置上にロードされている場合には、二次記憶装置上から当該永続オブジェクトをロードすることなく、主記憶装置上の永続オブジェクトを参照することができ、処理の効率が向上する。また、主記憶装置上にロー

ドされた永続オブジェクトと同じ永続オブジェクトを二次記憶装置からロードすることを回避できるために、1つの永続オブジェクトが主記憶装置上で複数の異なったデータを保持することを回避できる。

【0027】なお、位置識別子が設定されている永続オブジェクトとが主記憶装置上に初めてロードされる際にアドレス対応テーブルに位置識別子とアドレスとを対応させて格納するようにすれば、主記憶装置上の領域を効率よく使うことができ、処理効率を向上させることができる。

【0028】また、位置識別子と、当該位置識別子が示す永続オブジェクトを管理する管理単位（例えば、ページ）とを対応させておき、当該管理単位毎にアドレス対応テーブルを備えるようにすれば、位置識別子から迅速に主記憶装置上の永続オブジェクトを参照することができる。すなわち、位置識別子によって主記憶装置上の永続オブジェクトを参照する場合には、位置識別子によって永続オブジェクトが管理されている管理単位を把握できるために、把握した管理単位に関するアドレス対応テーブルのみを探索すれば、アドレスが取得でき、当該主記憶装置上の永続オブジェクトを参照することができる。

【0029】本発明の請求項3に係る永続オブジェクト管理方法は、請求項2に記載した永続オブジェクト管理方法において、オブジェクト識別子が設定された永続オブジェクトが主記憶装置上にロードされた場合に、当該永続オブジェクトが当該永続オブジェクトのオブジェクト識別子に対応する主記憶装置上のオブジェクトテーブル内のエントリのアドレスを保持することを特徴とする。

【0030】上記した請求項3に係る永続オブジェクト管理方法では、オブジェクト識別子が設定された永続オブジェクトが主記憶装置上にロードされた場合に、主記憶装置上の当該永続オブジェクト（メモリイメージ）は、主記憶装置上でのオブジェクト・テーブル内の当該永続オブジェクトのオブジェクト識別子に対応するエントリのアドレスを保持するために、主記憶装置上の永続オブジェクトの領域に、オブジェクト識別子や位置識別子の領域を別途に確保することなく、迅速に自己のオブジェクトテーブル内のエントリにアクセスすることができる。

【0031】本発明の請求項4に係る永続オブジェクト管理方法は、請求項1乃至請求項3のいずれか1項に記載の永続オブジェクト管理方法において、バックアップされた永続オブジェクトを二次記憶装置に再成する場合には、当該永続オブジェクトがバックアップされた際に設定されていたオブジェクト識別子を当該永続オブジェクトに再度設定することを特徴とする。

【0032】上記した請求項4に係る永続オブジェクト管理方法では、永続オブジェクトを再生する場合に、応

用プログラム等がバックアップ時に永続オブジェクトに設定されていたオブジェクト識別子を当該永続オブジェクトへ再度設定することができるために、永続オブジェクトを再生したオブジェクト指向データベースでの各オブジェクト間の参照一貫性をバックアップ時と同様に保つことができる。

#### 【0033】

【発明の実施の形態】本発明に係る永続オブジェクト管理方法を実施するオブジェクト指向データベースの一実施例を図1を参照して説明する。このオブジェクト指向データベースは、応用プログラム1が二次記憶装置2に格納された永続オブジェクト群を管理する目的で利用しており、永続オブジェクトを応用プログラム1が使用可能なメモリ形式として保持する主記憶装置としてのオブジェクトバッファ3と、永続オブジェクトが格納されているページを二次記憶装置2で格納されているディスク形式として保持するページバッファ4と、オブジェクトバッファ3と二次記憶装置2との間での永続オブジェクトの入出力管理を行う永続オブジェクト管理部5と、永続オブジェクト管理部5の管理に従ってページバッファ4と二次記憶装置2との間でのページの入出力処理を管理するページバッファ管理部6と、ページバッファ管理部6の管理に応じてページバッファ4と二次記憶装置2とのデータの入出力処理を行うディスク入出力部7と、応用プログラム1と二次記憶装置2と間でのデータのやりとりを制御する問い合わせ処理部8と、永続オブジェクト等といったデータの通信を制御する通信管理部9と、並行処理を管理するトランザクション管理部10と、を有している。

【0034】永続オブジェクト管理部5は、二次記憶装置2上のデータベースファイル内のデータを永続オブジェクト単位で管理しており、例えば、応用プログラム1が操作する対象の永続オブジェクトをページバッファ4からオブジェクトバッファ3へロードし、変更のあった永続オブジェクトを必要に応じてページバッファ4に書き戻す。なお、操作対象の永続オブジェクトがページバッファ4にない場合には、ページバッファ管理部6に、二次記憶装置2から操作対象の永続オブジェクトを読み出す指示を行う。

【0035】ページバッファ管理部6は、二次記憶装置2上のデータベースファイル内の永続オブジェクト等のデータをページ単位で管理し、例えば、永続オブジェクト管理部5から指示のあった永続オブジェクトをページ単位で二次記憶装置2上のデータベースファイルからページバッファ4へ読み出し、ページバッファ4上のページが変更された場合には当該ページを必要に応じて二次記憶装置2上のデータベースファイルへ書き戻す。

【0036】次に、本発明に係る永続オブジェクト管理方法を図2を参照して説明する。なお、図1と同様な部分には同一の番号を付し、重複する説明は省略する。二

次記憶装置2は、オブジェクト識別子が設定されている"オブジェクト1"と、位置識別子が設定されている"オブジェクト2"と、オブジェクトテーブル20とを有している。なお、永続オブジェクトを生成する方法は後述して説明する。

【0037】"オブジェクト1"は、自己を参照する関係と"オブジェクト2"を参照する関係とを有しており、自己の"オブジェクト識別子1"と"オブジェクト2"に設定されている"位置識別子2"とが保持されている。オブジェクトテーブル20には"オブジェクト識別子1"と、"オブジェクト1"の二次記憶装置2上の位置を示す"位置識別子1"とを対応付けて保持しており、位置識別子が設定されている"オブジェクト2"に関する情報は保持しない。

【0038】ページバッファ4は、二次記憶装置2に格納されている各永続オブジェクトやオブジェクトテーブルを格納しているページのうち、ページバッファ管理部6及びディスク入出力部7によって読み出されたページを保持している。この図2では、"オブジェクト1"を格納する"ページ1"、"オブジェクト2"を格納する"ページ2"、"オブジェクトテーブル20"を格納する"ページ3"が読み出された状態を表している。なお、ページバッファ4が保持している永続オブジェクト等は、二次記憶装置2上で格納されていたディスク形式データで構成されている。

【0039】オブジェクトバッファ3は、"オブジェクト1"と"オブジェクト2"と、オブジェクトテーブル30と、アドレス対応テーブル3aとを有している。"オブジェクト1"と"オブジェクト2"は、永続オブジェクト管理部5によってページバッファ4からロードされたものである。なお、ロードに際して、各永続オブジェクトはディスク形式データからメモリ形式データへ変換される。また、この際に、オブジェクト識別子が設定されている永続オブジェクトの場合には、永続オブジェクト管理部5がオブジェクトテーブル30のロードした永続オブジェクトに関するエントリのアドレスを当該永続オブジェクトに保持させる。

【0040】オブジェクトテーブル30は、"オブジェクト識別子1"と、"位置識別子1"と、"アドレス識別子1"とを有しており、"アドレス識別子1"は、"オブジェクト1"がオブジェクトバッファ3にロードされた際に永続オブジェクト管理部5によって設定され、"オブジェクト1"が配置されたアドレスを表している。

【0041】アドレス対応テーブル3aは、"位置識別子2"と、"アドレス識別子2"とを有しており、"オブジェクト2"に設定された"位置識別子2"と、"オブジェクト2"が配置されたオブジェクトバッファ3上のアドレスを示す"アドレス識別子2"との組は、"オブジェクト2"がオブジェクトバッファ3に初めてロー

ドされた際に、永続オブジェクト管理部5によって設定される。

【0042】次に、永続オブジェクトを生成する動作を図3を参照して説明する。永続オブジェクトを生成する動作は、応用プログラム1が永続オブジェクトの生成を要求した場合に、永続オブジェクト管理部5が以下に示す手順で実行する。まず、生成する永続オブジェクトを保持するための領域を二次記憶装置2上及びオブジェクトバッファ上に確保する(ステップS1、S2)。これにより、当該永続オブジェクトが保持される二次記憶装置2上の位置を表す位置識別子及び、オブジェクトバッファ3上での当該永続オブジェクトが配置されるアドレスを表すアドレス識別子が決定する。

【0043】次に、応用プログラム1の指示によって永続オブジェクトに設定する識別子の種類を検出する(ステップS3)。識別子が位置識別子である場合、すなわち、ユーザ或いは応用プログラムによって生成する永続オブジェクトの属性に基づく参照一貫性を保つためにかかる保守コストが所定値以上であると判断された場合には、当該オブジェクトを管理するページに関するアドレス対応テーブルに、既に決定した当該永続オブジェクトの位置識別子、及びアドレス識別子を登録する(ステップS4)。これによって、永続オブジェクトに内容を登録する等といった初期化が可能となる。

【0044】一方、識別子がオブジェクト識別子である場合、すなわち、ユーザ或いは応用プログラム1によって生成する永続オブジェクトの属性に基づく参照一貫性を保つためにかかる保守コストが所定値未満であると判断された場合には、設定するオブジェクト識別子が応用プログラム1によって指定されているか否かを検出し

(ステップS5)、指定されている場合は、予めバックアップした永続オブジェクトをリストアすることを表しているため、バックアップ時に永続オブジェクトが設定されていた識別子、すなわち指定されたオブジェクト識別子を保持する。一方、指定されていない場合には、一意値を生成し、当該一意値をオブジェクト識別子として保持する(ステップS6)。

【0045】次いで、二次記憶装置2上のオブジェクトテーブル20に、保持しているオブジェクト識別子と既に決定した位置識別子とを登録し(ステップS7)、オブジェクトバッファ3上のオブジェクトテーブル30に、保持しているオブジェクト識別子と既に決定した位置識別子及びアドレス識別子とを登録し(ステップS8)、オブジェクトバッファ3上のオブジェクトテーブル30に登録したオブジェクト識別子、位置識別子、アドレス識別子を保持するエントリのアドレスをオブジェクトバッファ3上の永続オブジェクトに保持させる(ステップS9)。これにより、永続オブジェクトに内容等を登録する初期化が可能となる。

【0046】初期化が可能になった場合には、応用プロ



グラム1の指示によってオブジェクトバッファ3上の永続オブジェクトを初期化し(ステップS10)、オブジェクトバッファ3上で初期化された内容を、ページバッファ4に管理されているページへ反映し(ステップS11)、ページバッファ4上のページを二次記憶装置2上のページへ反映させる(ステップS12)。

【0047】なお、上記した生成動作は、トランザクション処理として実行されるため、一連の処理の一部として、正常終了するか、もしくは異常終了するかは応用プログラム1によって決定される。異常終了する場合は、二次記憶装置2上並びにオブジェクトバッファ3上のすべての状態は、生成動作開始以前の状態に復帰される。

【0048】次に、生成した永続オブジェクトを参照する動作を説明するにあたって、まず、永続オブジェクトを参照する方法を図4を参照して説明する。図4(a)では、オブジェクト識別子による永続オブジェクトの参照方法を"オブジェクト1"が"オブジェクト2"を参照する場合を例にとって説明する。図4(a-1)は、参照先の"オブジェクト2"が二次記憶装置2上にある場合の参照方法であり、この参照方法はオブジェクトバッファ3上のオブジェクトテーブル3oを介して間接的に参照される。すなわち、参照先を示す"オブジェクト識別子2"に基づいてオブジェクトテーブル3oから当該"オブジェクト識別子2"に対応する"位置識別子2"を取得し、当該"位置識別子2"によって二次記憶装置2上の"オブジェクト2"を参照する。

【0049】図4(a-2)は、参照先の"オブジェクト2"がオブジェクトバッファ3上にある場合の参照方法であり、既にオブジェクトバッファ3へ"オブジェクト2"がロードされているのでオブジェクトテーブル3oには、当該"オブジェクト2"が配置されているアドレスを示す"アドレス識別子2"が保持されている。したがって、"オブジェクト識別子2"によって、オブジェクトテーブル3oを検索することによって、アドレス識別子2を得ることができ、"オブジェクト2"を参照することができる。また、一度"アドレス識別子2"を得た場合には、この"アドレス識別子2"を"オブジェクト1"自身が保持しておけば、当該"アドレス識別子2"で直接的に"オブジェクト2"を参照することが可能となる。

【0050】図4(b)では、位置識別子による永続オブジェクトの参照方法を"オブジェクト3"が"オブジェクト4"を参照する場合を例にとって説明する。図4(b-1)は、参照先の"オブジェクト4"が二次記憶装置2上にある場合の参照方法であり、この参照方法は直接的に参照される。すなわち、参照先を示す"位置識別子4"に基づいて二次記憶装置2上の"オブジェクト4"を参照する。

【0051】図4(b-2)は、参照先の"オブジェクト4"がオブジェクトバッファ3上にある場合の参照方

法であり、既にオブジェクトバッファ3へ"オブジェクト4"がロードされているのでページに関するアドレス対応テーブル3aには、当該"オブジェクト4"が配置されているアドレスを示すアドレス識別子4が保持されている。したがって、"位置識別子4"によって当該"位置識別子4"が含まれるページに関するアドレス対応テーブル3aを検索することによって、"アドレス識別子4"が得られ、迅速に"オブジェクト4"を参照することができる。また、一度"アドレス識別子4"を得た場合には、この"アドレス識別子4"をオブジェクトバッファ3上の"オブジェクト3"が保持しておけば、当該"アドレス識別子4"で直接的にオブジェクトバッファ3上の"オブジェクト4"を参照することが可能となる。

【0052】次に、上記した参照方法を行う動作を図5を参照して説明する。永続オブジェクトを参照する動作は、応用プログラム1が永続オブジェクトの参照を要求した場合に、永続オブジェクト管理部5によって以下に示す動作が行われる。まず、応用プログラム1からの参照要求において指定された識別子の種類によって後続処理を決定する(ステップS13)。これにより、参照元の永続オブジェクト若しくは応用プログラム1が保持している参照先の永続オブジェクトを示す識別子によって、以下に示す3つの動作のいずれかを行うこととなる。

【0053】アドレス識別子によって参照要求があった場合には、当該アドレス識別子の示す永続オブジェクトがオブジェクトバッファ3上で有効であるか否かを判定し、有効であれば、そのまま永続オブジェクトを参照する(ステップS14)。なお、オブジェクトバッファ3上の永続オブジェクトが無効であれば、二次記憶装置2上での永続オブジェクトのデータを該当するアドレスに再度ロードして参照する。

【0054】オブジェクト識別子によって参照要求があった場合には、オブジェクトバッファ3上のオブジェクトテーブル3oから当該オブジェクト識別子に対応する位置識別子とアドレス識別子とを探索し(ステップS15)、アドレス識別子が保持されているか否かを検出する(ステップS16)。なお、オブジェクトテーブル3o自体がオブジェクトバッファ3上に読み出されていないければ、当該オブジェクトテーブルを二次記憶装置2上からロードする。

【0055】そして、アドレス識別子がオブジェクトテーブル3oから得られた場合には、当該アドレス識別子によって永続オブジェクトを参照する(ステップS14)。一方、アドレス識別子が得られない場合には、永続オブジェクトが格納されている二次記憶装置2上のページを位置識別子から特定して読みだし、ページ内の永続オブジェクトをオブジェクトバッファ3上にロードする。そして、永続オブジェクトを読み出したオブジェク

トバッファ3上のアドレスをアドレス識別子としてオブジェクトテーブル30に登録する(ステップS17、S18)。

【0056】そして、オブジェクトテーブル30に保持された、読み出した永続オブジェクトに関するオブジェクト識別子、位置識別子、アドレス識別子の組のエントリのアドレスを、読み出された永続オブジェクトに保持させる(ステップS19)。次いで、アドレス識別子によって永続オブジェクトを参照する(ステップS14)。上記した永続オブジェクトが保持するエントリのアドレスによって、永続オブジェクトは自己の永続オブジェクトに関する識別子をオブジェクトテーブル30内から迅速に得ることができる。

【0057】位置識別子によって参照指示があった場合には、位置識別子から参照する永続オブジェクトを格納しているページを特定し、当該ページに関するアドレス対応テーブル3aから位置識別子に対応するアドレス識別子を探索し、アドレス識別子が保持されているか否かを検出する(ステップS20、S21)。アドレス識別子が保持されている場合には、当該アドレス識別子によって永続オブジェクトを参照する(ステップS14)。一方、アドレス識別子が保持されていない場合には、ページバッファ4上のページから参照先の永続オブジェクトをオブジェクトバッファ3上にロードする。なお、該当するページがページバッファ上に読み出されていない場合には、二次記憶装置2からロードする。

【0058】次いで、オブジェクトバッファ上にロードした永続オブジェクトのアドレス識別子を、当該永続オブジェクトを管理するページに関するアドレス対応テーブル3aに登録する(ステップS22、S23)。そして、前記アドレス識別子によって永続オブジェクトを参照する(ステップS13)。

【0059】上記説明したように、永続オブジェクトに位置識別子または、オブジェクト識別子のいずれを設定するかを、永続オブジェクトの属性に基づいて参照一貫性を保つためにかかる保守コストに基づいて設定することができる。また、永続オブジェクトがオブジェクトバッファ上にロードされている場合には、当該永続オブジェクトの配置されているアドレスと当該永続オブジェクトに設定されている識別子とが管理されているために、オブジェクトバッファ上に配置されている当該永続オブジェクトを参照することができ、新たに二次記憶装置から読み出すといった重複したロードを回避できるとともに、迅速に参照することができる。

【0060】また、オブジェクト識別子が設定された永続オブジェクトをオブジェクトバッファ上にロードする際に、当該永続オブジェクトに関するオブジェクトテーブル上のエントリのアドレスを前記永続オブジェクトが保持するようにしたために、後に、当該永続オブジェクトが自己を参照する場合には、保持しているアドレスに

よって、オブジェクトテーブル内から自己に関する識別子群を迅速に得ることができる。また、バックアップした永続オブジェクトをリストアさせる場合において、永続オブジェクトに設定するオブジェクト識別子として、バックアップ時に永続オブジェクトに設定されていたオブジェクト識別子を設定することができる。

【0061】なお、上記した実施例では、アドレス対応テーブルはページ毎に永続オブジェクトの位置識別子とアドレスとを管理して、位置識別子に基づいてアドレス識別子を探索する範囲を減らして探索効率を向上していたが、上記に限らず、一つのアドレス対応テーブルで全ての位置識別子を持った永続オブジェクトに関して管理するようにしてもよい。

【0062】

【発明の効果】以上説明したように本発明に係る永続オブジェクト管理方法では、永続オブジェクトの属性に基づく参照一貫性を保つためにかかる保守コストに基づいて、永続オブジェクトに位置識別子或いはオブジェクト識別子のいずれかを設定するようにしたために、保守コストを低下することができるとともに、参照効率も向上することができる。

【0063】また、本発明に係る永続オブジェクト管理方法では、永続オブジェクトを主記憶装置上にロードする際に、当該永続オブジェクトの配置されているアドレスと当該永続オブジェクトに設定されている識別子とを管理するようにしたために、主記憶装置上に配置されている当該永続オブジェクトを迅速に参照することができ、新たに二次記憶装置から読み出すといった重複したロードを回避することができ、主記憶装置を有効に利用することができる。

【0064】また、本発明に係る永続オブジェクト管理方法では、オブジェクト識別子が設定された永続オブジェクトを主記憶装置上にロードする際に、当該永続オブジェクトに関するオブジェクトテーブルのエントリのアドレスを前記永続オブジェクトに保持させるようにしたために、後に、当該永続オブジェクトが自己を参照する場合に、保持しているアドレスによって、オブジェクトテーブル内から自己に関する識別子群を迅速に得ることができ、処理効率が向上する。

【0065】また、本発明に係る永続オブジェクト管理方法では、永続オブジェクトに設定するオブジェクト識別子として、バックアップ時に永続オブジェクトに設定されていたオブジェクト識別子を設定できるようにしたために、当該永続オブジェクトをリストアした後のオブジェクト指向データベース内の各永続オブジェクト間の参照一貫性をバックアップした際と同様に再現することができる。

【図面の簡単な説明】

【図1】 本発明に係る永続オブジェクト管理方法を実施するオブジェクト指向データベースの構成図である。

【図2】 本発明に係る永続オブジェクト管理方法を説明する構成図である。

【図3】 永続オブジェクトの生成動作を説明するフローチャートである。

【図4】 永続オブジェクトの参照方法を説明する図である。

【図5】 永続オブジェクトを参照する手順を説明するフローチャートである。

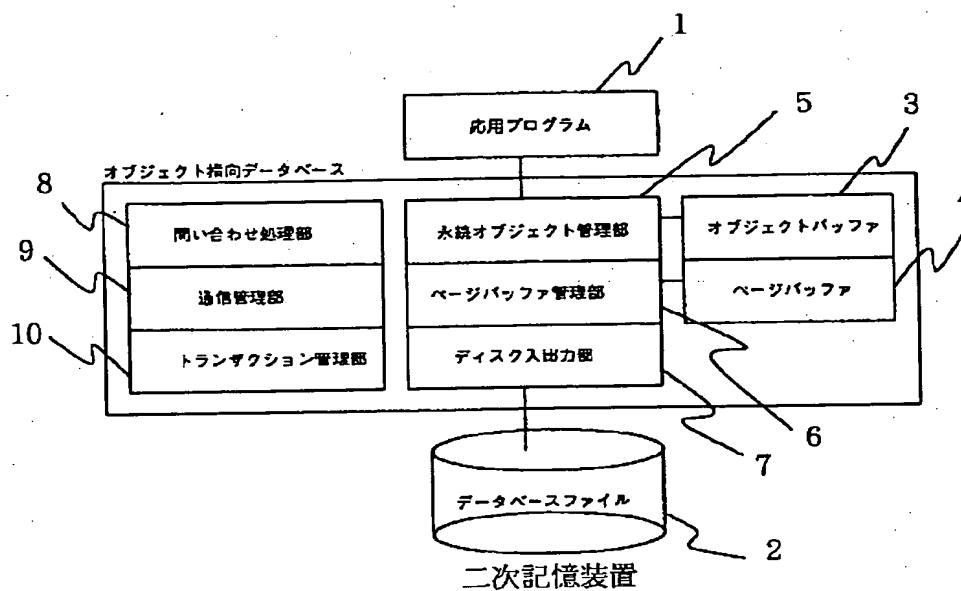
【図6】 従来例に係るオブジェクト識別子のみで永続

オブジェクトを管理するオブジェクト指向データベースの永続オブジェクト管理方法を説明する図である。

【符号の説明】

1・・・応用プログラム、 2・・・二次記憶装置、 3・・・オブジェクトバッファ、 3a・・・オブジェクトテーブル、 3b・・・アドレス対応テーブル、 4・・・ページバッファ、 5・・・永続オブジェクト管理部、 6・・・ページバッファ管理部、 7・・・ディスク入出力部、

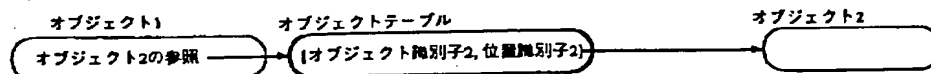
【図1】



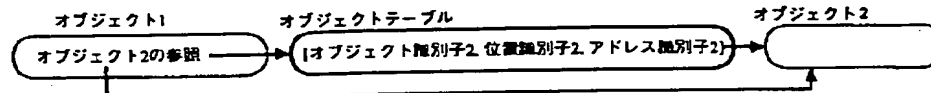
【図4】

(a) オブジェクト識別子によるオブジェクトの参照

(a-1) 二次記憶装置上

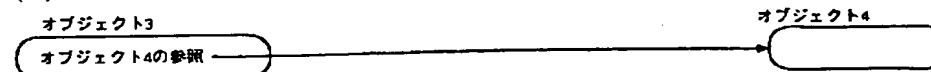


(a-2) 主記憶装置上

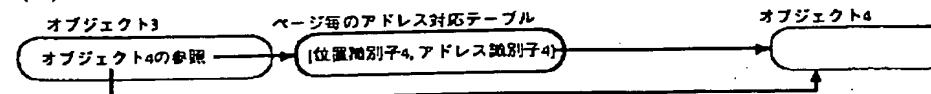


(b) 位置識別子によるオブジェクトの参照

(b-1) 二次記憶装置上



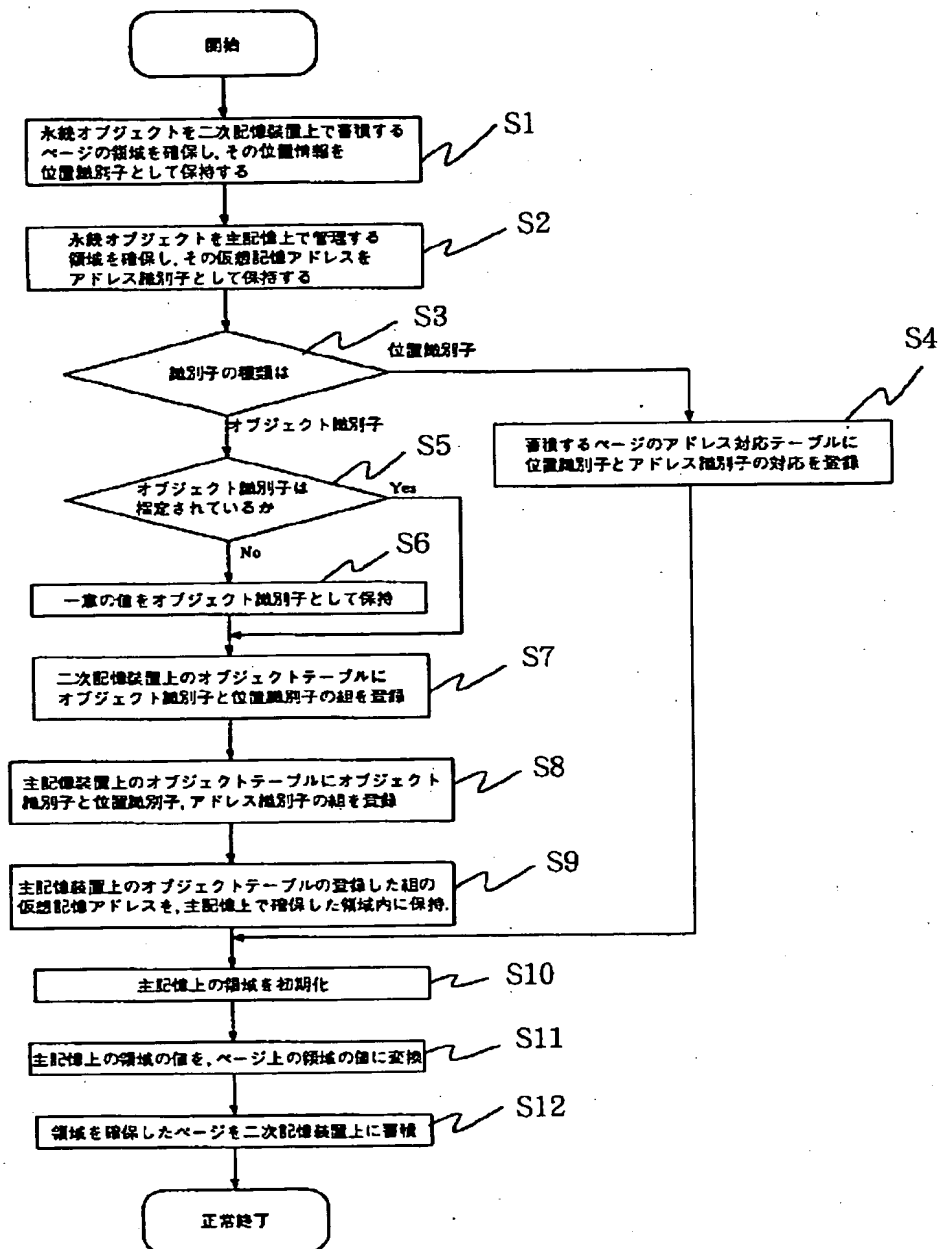
(b-2) 主記憶装置上



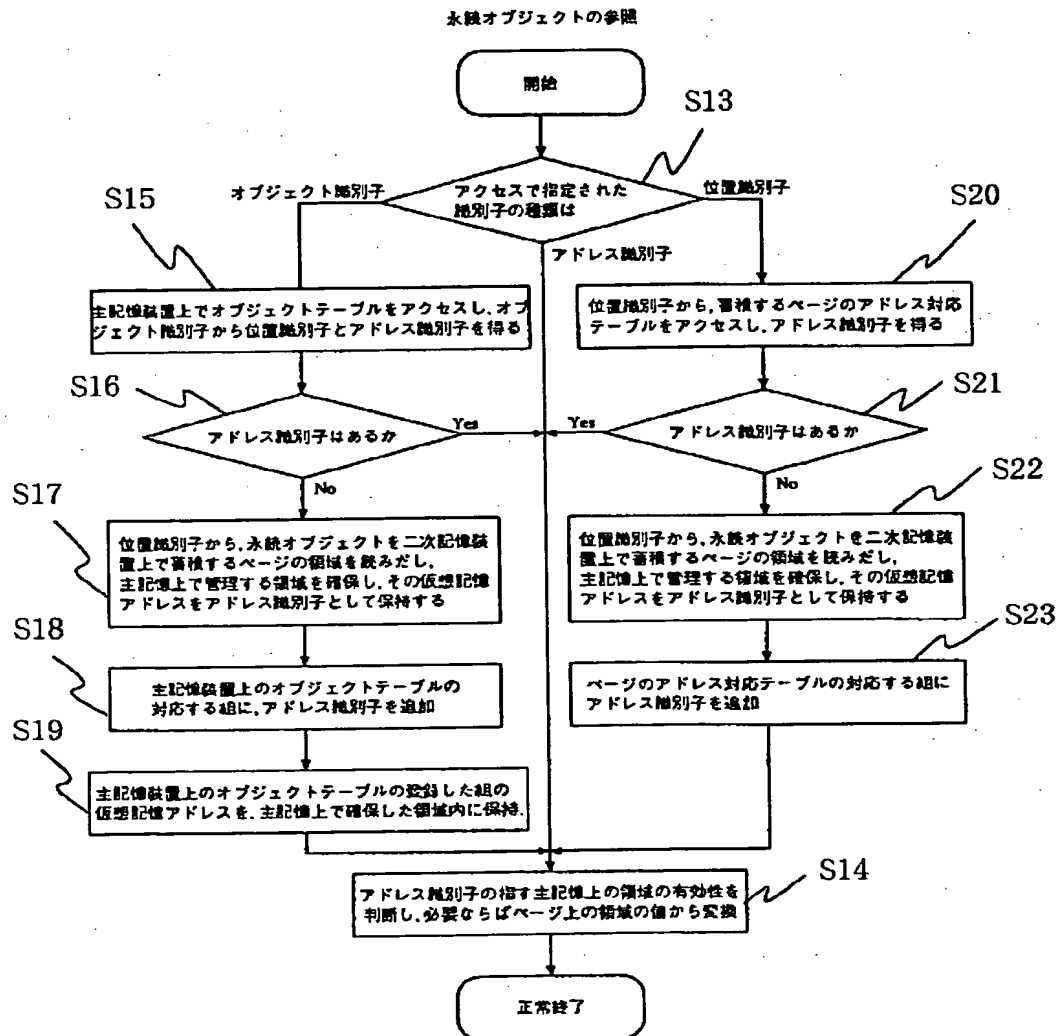


【図3】

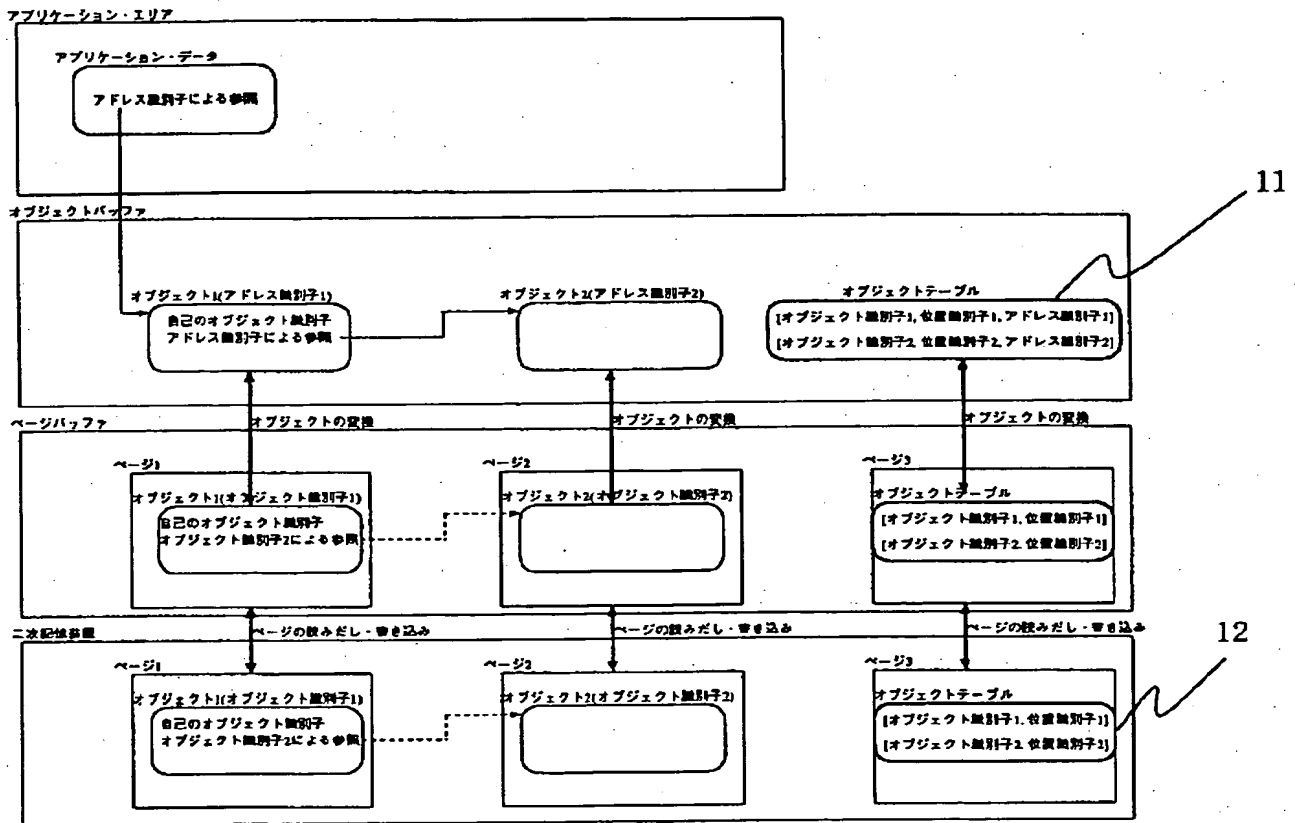
## 永続オブジェクトの生成



【図5】



【図6】



**\* NOTICES \***

JPO and NCIPi are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

**CLAIMS**

[Claim(s)]

[Claim 1] The permanent object management method characterized by to set up the object identifier of the logical value of a meaning based on the maintenance cost applied in order to maintain the referential integrity based on the attribute of the permanent object concerned, in case a permanent object is generated being beyond a predetermined value, and to set up the location identifier which shows the storing location in the secondary storage in which the object concerned is stored based on a maintenance cost being under a predetermined value in the object oriented database which identifies and refers to a permanent object by the identifier.

[Claim 2] In case the permanent object to which the object identifier is set is loaded on main storage in the permanent object management method indicated to claim 1 While the address on the main storage with which the permanent object concerned is loaded is matched with an object identifier and storing in the object table on main storage In case the permanent object to which the location identifier is set is loaded on main storage Match with a location identifier the address on the main storage with which the permanent object concerned is loaded, and it stores in the address-mapping table on main storage. In case an object identifier refers a permanent object When it searches for an object table and there is the address corresponding to the object identifier concerned, the address concerned refers the permanent object on main storage. The permanent object management method characterized by referring to the permanent object on main storage with the address concerned when it looks for an address-mapping table and there is the address corresponding to the location identifier concerned, in case a location identifier refers a permanent object.

[Claim 3] The permanent object management method characterized by the permanent object concerned on main storage holding the address of the entry in the object table on the main storage corresponding to the object identifier of the permanent object concerned when the permanent object to which the object identifier was set is loaded on main storage in the permanent object management method indicated to claim 2.

[Claim 4] The permanent object management method characterized by setting again the object identifier set up when the permanent object concerned was backed up as the permanent object concerned in re-\*\*(ing) the permanent object backed up by any 1 term of claim 1 thru/or claim 3 in the permanent object management method of a publication to a secondary storage.

[Translation done.]

**\* NOTICES \***

JPO and NCIPi are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.



## DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention] This invention relates to the management method of permanent objects, such as an approach of identifying a permanent object, and a method of referring to a permanent object, in an object oriented database.

[0002]

[Description of the Prior Art] The object oriented database which manages the object which doubled the procedure which operates data and data conventionally can be considered, and since it is efficiently manageable while being able to model easily the data complicated and diversified, it is observed especially in the field which manages data, such as a document and multimedia data.

[0003] One of the descriptions of this object oriented database The identifier which assigned the reference relation between the objects which show that the object of 1 is referring to other objects to each object expresses. It is being able to follow from the object of 1 to other objects by the assigned identifier concerned. As the method of a reference-related expression For example, in expressing the reference to the object of a reference place of a referring to agency from an object, it is expressing by holding the identifier of the object of a reference place to the member variable in the object of a referring to agency etc.

[0004] What is necessary is just not to merely only assign the above-mentioned identifier suitably, and it must secure the referential integrity that an object is identified certainly and can be referred to. That is, the situation that the object of assigning and keeping the same identifier to different OBUJETO etc. is not discriminable must be avoided.

[0005] Epicritic [ of identifying the above-mentioned object certainly ] is indicated by 85-86 pages of Addison-Wesley Publishing R.G.G.Cattel work "Object Data Management Revised Edition" (1994) (henceforth reference 1), and 151-158 pages.

[0006] As one of the identifiers assigned to a permanent object, there is a method using the location identifier decided by the storing location on the secondary storage of a permanent object, and by this method, the location identifier of the permanent object of a reference place can be held, and by the location identifier concerned, the permanent object of a referring to agency can pinpoint immediately the location on the secondary storage of the object of a reference place, and refer to it for it.

[0007] Moreover, there is a method using the object identifier of the logical value of a meaning as other identifiers assigned to a permanent object, and refer to the object of a reference place for the object of a referring to agency by this method by holding the object table 11 which manages the group of the location on the secondary storage of a permanent object, and the object identifier assigned to the permanent object as shown in drawing 6 on main storage (on an object buffer). That is, it searches for an object table 11, the location identifier corresponding to the object identifier showing the permanent object of the reference place which the permanent object of a referring to agency holds is acquired from an object table 11, and things can be carried out with reference to the permanent object of the reference place on a secondary storage by the location identifier concerned. In addition, the above-mentioned object table 11 is usually stored as an object table 12 perpetuated on the secondary storage.

[0008] For example, when "an object 1" refers to the "object 2" to which "the object identifier 2" was assigned, it sets. "The object 1" holds the "object identifier 2" assigned to "the object 2." The "location identifier 2" corresponding to the "object identifier 2" concerned is acquired from the object table 11 on main storage, and it becomes possible to refer to the "object 2" on a secondary storage by the acquired "location identifier 2."

[0009] Moreover, in the object oriented database, load a permanent object to main storage from a secondary storage, and read-out actuation and write-in actuation are performed. for example, when two or more permanent objects refer to one certain permanent object, or when one permanent object carries out multiple-times reference of one certain permanent object The situation which overlaps and loads the same permanent object had occurred, and decline in processing effectiveness and the said problem to which a permanent object with different data exists on two or more main storage were also produced by such duplicate loading.

[0010] The approach of managing the address identifier which expresses the address of the permanent object once loaded on main storage with invention indicated by JP,5-225034,A on an object table, the approach the permanent object of the reference origin on main storage holds the address identifier of the permanent object of a reference place, and \*\* are carried out to this problem. In this invention, while avoiding loading of the permanent object which overlapped by registering the address identifier when 3 groups of an object identifier, a location identifier, and an address identifier are managed as an object table 11 and the permanent object is loaded on main storage as shown in drawing 6 , and referring to the permanent object on main storage by the address identifier concerned and guaranteeing referential integrity, the effectiveness of reference processing of a permanent object is improved.

[0011] Moreover, in this invention, it constitutes from a group of the volatilization object generated by programming language in the structure (henceforth a memory image) of the permanent object loaded on main storage, and a handler with the information on permanent object management, and reference to the permanent object on main storage etc. is operated by the handler.

[0012] Moreover, J.E.Moss work "Working with Persistent Objects : in To Swizzle or Notto Swizzle"IEEE Transaction on Software Engineering Vol.18 No.8, the conversion method about the address identifier for expressing reference by the memory image (1992) is indicated.

[0013]

[Problem(s) to be Solved by the Invention] In the object oriented database only using the conventional location identifier, in case a permanent object is referred to, the permanent object of the reference place on a secondary storage can be immediately specified and referred to from a location identifier, and the effectiveness of reference is good. However, in the object oriented database using this location identifier, when changing the location of the permanent object on a secondary storage, or when deleting the permanent object itself, it will be necessary to change or delete the location identifier which is referring to the permanent object changed or deleted and which a permanent object holds altogether, and will take time and effort.

[0014] On the other hand, it is not necessary to change the object identifier which expresses with the object oriented database only using an object identifier the reference place currently held at the permanent object of a referring to agency when the location of a permanent object changes, or when the permanent object itself deletes, and does not take time and effort that what is necessary is just to change correspondence with the object identifier and location identifier which are managed by the object table.

[0015] However, in the object oriented database only using an object identifier, it will be necessary to manage the location on the secondary storage of the object to which the object identifier and the object identifier concerned were assigned as an object table, if the number of permanent objects increases, an object table will become large, and many main storage tops or storage regions of a secondary storage will be needed. Moreover, as an object table becomes large, retrieval cost called the effectiveness or time amount of the retrieval performed in the case of reference becomes larger. Furthermore, when a multiple user refers to a permanent object to coincidence, retrieval by the object table will carry out a conflict, the

parallelism of processing will fall, and processing effectiveness will fall as a result.

[0016] Generally, in the object oriented database, while avoiding loading of a up to [ the main storage of the duplicate permanent object ] and guaranteeing the referential integrity to a permanent object, to improve the effectiveness of reference processing of a permanent object is desired.

[0017] Moreover, it is necessary to back up a specific permanent object to tertiary storages, such as a tape, and to make it possible to restore from tertiary storage to a secondary storage after that from the database file on a secondary storage, for maintenance of a permanent object, and the reference to the permanent object restored after backup needs to be maintained like the time of backup in an object oriented database.

[0018] Then, this invention was made in view of the above-mentioned conventional situation, and aims at offering the permanent object management method which can be made to be able to demonstrate both advantages and can improve processing effectiveness by using properly the permanent object management method performed using a location identifier, and the permanent object management method performed using an object identifier.

[0019] Moreover, while this invention avoids loading of a up to [ the main storage of the duplicate permanent object ] in the object oriented database which uses the above-mentioned object identifier and the above-mentioned location identifier properly and maintains the referential integrity to a permanent object, it aims at offering the permanent object management method which can raise processing effectiveness. Moreover, this invention aims at offering the permanent object management method which can improve integrity in the object oriented database which uses the above-mentioned object identifier and the above-mentioned location identifier properly.

[0020]

[Means for Solving the Problem] It is what was made paying attention to the maintenance cost applied in order that this invention may maintain referential integrity based on the attribute of a permanent object, in order to attain the above-mentioned purpose. When a maintenance cost is beyond a predetermined value, set up the logical object identifier of a meaning to a permanent object, and a maintenance cost is held down as much as possible. When a maintenance cost is under a predetermined value, the location identifier which shows the storing location on a secondary storage to a permanent object is set up, and the reference effectiveness of a permanent object is improved.

[0021] In case the maintenance cost applied here in order to maintain referential integrity based on the attribute of a permanent object performs processing which moves the location on the secondary storage of a permanent object, processing which deletes a permanent object from on a secondary storage, it means cost called the time and effort which it takes in order to maintain without breaking down the referential integrity between each permanent object before and after processing. For this reason, the permanent object referred to from many permanent objects, for example has a high maintenance cost, and the maintenance cost of the permanent object referred to only from the permanent object of one or a fraction is low. Moreover, once creating, the maintenance cost of the permanent object of not being moved or deleted is low, and the permanent object which may change a storing location frequently can say that a maintenance cost is high. In addition, a predetermined value can be arbitrary, and can be good, and the user of an object oriented database can also set it up, and it can also set up an application program.

[0022] Then, the permanent object management method concerning claim 1 of this invention In the object oriented database which identifies and refers to a permanent object by the identifier The object identifier of the logical value of a meaning is set up based on the maintenance cost applied in order to maintain the referential integrity based on the attribute of the permanent object concerned being beyond a predetermined value, in case a permanent

object is generated. It is characterized by setting up the location identifier which shows the storing location in the secondary storage in which the object concerned is stored based on a maintenance cost being under a predetermined value.

[0023] In the permanent object management method indicated to above-mentioned claim 1, an object identifier and a location identifier can be set as a permanent object with the maintenance cost applied in order to maintain the referential integrity based on the attribute of a permanent object. Therefore, since the number of permanent objects with which the maintenance cost applied in order to maintain referential integrity in an object oriented database database can be reduced, and an object identifier is set up can also be stopped, while being able to reduce the magnitude of an object table and being able to use effectively the field on main storage and a secondary storage, in case an object identifier refers a permanent object, the retrieval cost in an object table can be reduced.

[0024] The permanent object management method concerning claim 2 of this invention In case the permanent object to which the object identifier is set is loaded on main storage in the permanent object management method indicated to claim 1 While the address on the main storage with which the permanent object concerned is loaded is matched with an object identifier and storing in the object table on main storage In case the permanent object to which the location identifier is set is loaded on main storage Match with a location identifier the address on the main storage with which the permanent object concerned is loaded, and it stores in the address-mapping table on main storage. In case an object identifier refers a permanent object When it searches for an object table and there is the address corresponding to the object identifier concerned, the address concerned refers the permanent object on main storage. In case a location identifier refers a permanent object, when it looks for an address-mapping table and there is the address corresponding to the location identifier concerned, it is characterized by referring to the permanent object on main storage with the address concerned.

[0025] In the permanent object management method concerning above-mentioned claim 2 In case the permanent object to which the object identifier already loaded on main storage was set is referred to By the object identifier which shows a reference place, search for an object table and the address is acquired. While the address concerned refers the permanent object on main storage, in case the permanent object to which the location identifier already loaded on main storage was set is referred to The location identifier which shows a reference place searches for an address-mapping table, the address is acquired, and the address concerned refers the permanent object on main storage.

[0026] Therefore, without loading the permanent object concerned from on a secondary storage, when the permanent object to refer to is loaded on main storage, the permanent object on main storage can be referred to, and the effectiveness of processing improves. Moreover, since it is avoidable to load the same permanent object as the permanent object loaded on main storage from a secondary storage, one permanent object can avoid holding the data with which plurality differed on main storage.

[0027] In addition, if a location identifier and the address are made to correspond to an address-mapping table and it is made to store in it in case the permanent object to which the location identifier is set is loaded for the first time on main storage, the field on main storage can be used efficiently and processing effectiveness can be raised.

[0028] Moreover, if the location identifier and the management unit (for example, page) which manages the permanent object which the location identifier concerned shows are made to correspond and it has an address-mapping table for every management unit concerned, the permanent object on main storage can be quickly referred to from a location identifier. That is, since the management unit in which the permanent object is managed by the location identifier can be grasped when a location identifier refers the permanent object on main

storage, if it looks only for the address-mapping table about the grasped management unit, the address can be acquired and the permanent object on the main storage concerned can be referred to.

[0029] In the permanent object management method indicated to claim 2, the permanent object management method concerning claim 3 of this invention is characterized by the permanent object concerned holding the address of the entry in the object table on the main storage corresponding to the object identifier of the permanent object concerned, when the permanent object to which the object identifier was set is loaded on main storage.

[0030] In the permanent object management method concerning above-mentioned claim 3 When the permanent object to which the object identifier was set is loaded on main storage, the permanent object concerned on main storage (memory image) In order to hold the address of an entry with which the object identifier of the permanent object concerned in the object table on main storage corresponds The field of the permanent object on main storage can be accessed quickly at the entry in a self object table, without securing the field of an object identifier or a location identifier separately.

[0031] The permanent object management method concerning claim 4 of this invention is characterized by setting again the object identifier set up when the permanent object concerned was backed up as the permanent object concerned, when re-\*\*(ing) the permanent object backed up by any 1 term of claim 1 thru/or claim 3 in the permanent object management method of a publication to a secondary storage.

[0032] In the permanent object management method concerning above-mentioned claim 4, since the object identifier by which the application program etc. was set as the permanent object at the time of backup can be again set to the permanent object concerned when reproducing a permanent object, the referential integrity between each object in the object oriented database which reproduced the permanent object can be kept being the same as that of the time of backup.

[0033]

[Embodiment of the Invention] One example of the object oriented database which enforces the permanent object management method concerning this invention is explained with reference to drawing 1 . This object oriented database is used in order to manage the permanent object group by which the application program 1 was stored in the secondary storage 2. The object buffer 3 as main storage with which an application program 1 holds a permanent object as an usable memory format, The page buffer 4 held as a disk format in which the page in which the permanent object is stored is stored by the secondary storage 2, With the permanent object Management Department 5 which performs input/output management of the permanent object between the object buffer 3 and a secondary storage 2 With the page buffer Management Department 6 which manages the radial transfer of the page between a page buffer 4 and a secondary storage 2 according to management of the permanent object Management Department 5 The disk input/output section 7 which performs radial transfer of the data of a page buffer 4 and a secondary storage 2 according to management of the page buffer Management Department 6, It has an application program 1, a secondary storage 2, the inquiry processing section 8 that controls an exchange of the data of a between, the communication management section 9 which controls the communication link of data called a permanent object etc., and the transaction management section 10 which manages parallel processing.

[0034] The permanent object Management Department 5 loads the permanent object of the object which has managed the data in the database file on a secondary storage 2 per permanent object, for example, an application program 1 operates to the object buffer 3 from a page buffer 4, and returns a permanent object with modification to a page buffer 4 if needed. In addition, when there is no permanent object for actuation in a page buffer 4, the directions

which read the permanent object for actuation from a secondary storage 2 are given to the page buffer Management Department 6.

[0035] The page buffer Management Department 6 returns the page concerned to the database file on a secondary storage 2 if needed, when the page on page buffer 4 HE read-out and a page buffer 4 is changed from the database file on a secondary storage 2 per page in the permanent object which managed data, such as a permanent object in the database file on a secondary storage 2, per page, for example, had directions from the permanent object Management Department 5.

[0036] Next, the permanent object management method concerning this invention is explained with reference to drawing 2. In addition, the same number is given to the same part as drawing 1, and the overlapping explanation is omitted. The secondary storage 2 has the "object 1" to which the object identifier is set, the "object 2" to which the location identifier is set, and object table 2o. In addition, the approach of generating a permanent object is mentioned later and explained.

[0037] "The object 1" has the relation which refers to self, and the relation which refers to "an object 2", and the "object identifier 1" of self and the "location identifier 2" set as "the object 2" are held. To object table 2o, "the object identifier 1" and the "location identifier 1" which shows the location on the secondary storage 2 of "an object 1" are matched and held, and the information about the "object 2" to which the location identifier is set is not held.

[0038] The page buffer 4 holds the page read by the page buffer Management Department 6 and the disk input/output section 7 among the pages which store each permanent object stored in the secondary storage 2, and an object table. The condition that the "page 1" which stores "an object 1", the "page 2" which stores "an object 2", and the "page 3" which stores "object table 2o" were read is expressed with this drawing 2. In addition, the permanent object which the page buffer 4 holds consists of disk format data stored on the secondary storage 2.

[0039] The object buffer 3 has "an object 1", "an object 2", object table 3o, and address-mapping table 3a. "An object 1" and "an object 2" are loaded by the permanent object Management Department 5 from a page buffer 4. In addition, on the occasion of a \*\* load, each permanent object is changed into memory formal data from disk format data. Moreover, when it is the permanent object by which the object identifier is set up in this case, the permanent object Management Department 5 makes the address of the entry about the permanent object which object table 3o loaded hold to the permanent object concerned.

[0040] Object table 3o has "the object identifier 1", and "the location identifier 1" and the "address identifier 1", and "the address identifier 1" is set up by the permanent object Management Department 5 when "an object 1" is loaded to the object buffer 3, and it expresses the address with which "the object 1" has been arranged.

[0041] Address-mapping table 3a has "the location identifier 2" and the "address identifier 2", and the group of the "location identifier 2" set as "the object 2" and the "address identifier 2" which shows the address on the object buffer 3 with which "the object 2" has been arranged is set up by the permanent object Management Department 5 when "an object 2" is loaded to the object buffer 3 for the first time.

[0042] Next, the actuation which generates a permanent object is explained with reference to drawing 3. Actuation which generates a permanent object is performed in the procedure which the permanent object Management Department 5 shows below, when an application program 1 requires generation of a permanent object. First, the field for holding the permanent object to generate is secured on a secondary storage 2 and an object buffer (steps S1 and S2). The location identifier which expresses by this the location on the secondary storage 2 by which the permanent object concerned is held, and the address identifier showing the address with which the permanent object concerned on the object buffer 3 is arranged are determined.

[0043] Next, the class of identifier set as a permanent object with directions of an application program 1 is detected (step S3). When an identifier is a location identifier (i.e., when the maintenance cost applied in order to maintain the referential integrity based on the attribute of the permanent object generated with a user or an application program is judged to be beyond a predetermined value), the location identifier and address identifier of the permanent object concerned already determined as the address-mapping table about the page which manages the object concerned are registered (step S4). By this, initialization of registering the contents into a permanent object etc. is attained.

[0044] On the other hand, when an identifier is an object identifier (i.e., when the maintenance cost applied in order to maintain the referential integrity based on the attribute of the permanent object generated with a user or an application program 1 is judged to be under a predetermined value) Detect (step S5), and when specified, whether the object identifier to set up is specified by the application program 1. Since it means restoring the permanent object which backed up beforehand, the identifier by which the permanent object was set up at the time of backup, i.e., the specified object identifier, is held. On the other hand, when not specified, a meaning value is generated and the meaning value concerned is held as an object identifier (step S6).

[0045] Subsequently, the location identifier already determined as the object identifier currently held is registered into object table 2o on a secondary storage 2 (step S7). The location identifier and address identifier which were already determined as the object identifier currently held are registered into object table 3o on the object buffer 3 (step S8). The address of the entry holding the object identifier registered into object table 3o on the object buffer 3, a location identifier, and an address identifier is made to hold to the permanent object on the object buffer 3 (step S9). Thereby, initialization which registers the contents etc. into a permanent object is attained.

[0046] When initialization becomes possible, the contents which initialized the permanent object on the object buffer 3 (step S10), and were initialized on the object buffer 3 by directions of an application program 1 are reflected to the page managed by the page buffer 4 (step S11), and the page on a page buffer 4 is made to reflect to the page on a secondary storage 2 (step S12).

[0047] In addition, since the above-mentioned generation actuation is performed as transaction processing, it is determined by the application program 1 as a part of a series of processings whether it terminates normally or it terminates abnormally. When terminating abnormally, all the conditions on the object buffer 3 return to the condition before generation actuation initiation at a secondary-storage 2 top list.

[0048] Next, in explaining the actuation which refers to the generated permanent object, how to refer to a permanent object is first explained with reference to drawing 4. Drawing 4 (a) explains the reference approach of the permanent object by the object identifier taking the case of the case where "an object 1" refers to "an object 2." Drawing 4 (a-1) is the reference approach in case the "object 2" of a reference place is on a secondary storage 2, and this reference approach is indirectly referred to through object table 3o on the object buffer 3. That is, based on the "object identifier 2" which shows a reference place, the "location identifier 2" corresponding to the "object identifier 2" concerned is acquired from object table 3o, and the "location identifier 2" concerned refers the "object 2" on a secondary storage 2.

[0049] Drawing 4 (a-2) is the reference approach in case the "object 2" of a reference place is on the object buffer 3, and since "the object 2" is already loaded to the object buffer 3, the "address identifier 2" which shows the address with which the "object 2" concerned is arranged is held at object table 3o. Therefore, by "the object identifier 2", by searching object table 3o, the address identifier 2 can be obtained and "an object 2" can be referred to. Moreover, if the "object 1" itself holds this the "address identifier 2" when "the address

identifier 2" is obtained once, it will become possible to refer to "an object 2" directly by the "address identifier 2" concerned.

[0050] Drawing 4 (b) explains the reference approach of the permanent object by the location identifier taking the case of the case where "an object 3" refers to "an object 4." Drawing 4 (b-1) is the reference approach in case the "object 4" of a reference place is on a secondary storage 2, and this reference approach is referred to directly. That is, based on the "location identifier 4" which shows a reference place, the "object 4" on a secondary storage 2 is referred to.

[0051] Drawing 4 (b-2) is the reference approach in case the "object 4" of a reference place is on the object buffer 3, and since "the object 4" is already loaded to the object buffer 3, the address identifier 4 which shows the address with which the "object 4" concerned is arranged is held at address-mapping table 3a about a page. Therefore, by searching address-mapping table 3a about the page in which the "location identifier 4" concerned is contained by "the location identifier 4", "the address identifier 4" is obtained and "an object 4" can be referred to quickly. Moreover, if the "object 3" on the object buffer 3 holds this the "address identifier 4" when "the address identifier 4" is obtained once, it will become possible to refer to the "object 4" on the object buffer 3 directly by the "address identifier 4" concerned.

[0052] Next, the actuation which performs the above-mentioned reference approach is explained with reference to drawing 5. Actuation as which the actuation which refers to a permanent object indicates it below by the permanent object Management Department 5 when an application program 1 requires reference of a permanent object is performed. First, the class of identifier specified in the reference demand from an application program 1 determines consecutiveness processing (step S13). The identifier which shows by this the permanent object of the reference place which the permanent object or application program 1 of a referring to agency holds will perform either of three actuation shown below.

[0053] When there is a reference demand by the address identifier, it judges whether the permanent object which the address identifier concerned shows is effective on the object buffer 3, and if effective, a permanent object will be referred to as it is (step S14). In addition, if the permanent object on an object buffer is invalid, the data of the permanent object on a secondary storage 2 will be again loaded to the corresponding address, and will be referred to.

[0054] It detects whether when there is a reference demand, it searches for the location identifier and address identifier corresponding to the object identifier concerned from object table 3o on the object buffer 3 (step S15), and the address identifier is held by the object identifier (step S16). In addition, if the object table 3o itself is not read on the object buffer 3, the object table concerned is loaded from a secondary storage.

[0055] And when an address identifier is obtained from object table 3o, the address identifier concerned refers a permanent object (step S14). On the other hand, when an address identifier is not obtained, the page on the secondary storage 2 in which the permanent object is stored is specified and read from a location identifier, and the permanent object in a page is loaded on the object buffer 3. And it registers with object table 3o by making into an address identifier the address on the object buffer 3 which read the permanent object (steps S17 and S18).

[0056] And the address of the entry of the group of the object identifier about the read permanent object held at object table 3o, a location identifier, and an address identifier is made to hold to the read permanent object (step S19). Subsequently, an address identifier refers a permanent object (step S14). With the address of the entry which the above-mentioned permanent object holds, a permanent object can obtain the identifier about a self permanent object quickly from the inside of object table 3o.

[0057] It detects whether when there are reference directions, the page which stores the permanent object referred to from a location identifier is specified, it searches for the address identifier corresponding to a location identifier from address-mapping table 3a about the page



concerned, and the address identifier is held by the location identifier (steps S20 and S21). When the address identifier is held, the address identifier concerned refers a permanent object (step S14). On the other hand, when the address identifier is not held, the permanent object of a reference place is loaded on the object buffer 3 from the page on a page buffer 4. In addition, when the corresponding page is not read on the page buffer, it loads from a secondary storage 2.

[0058] Subsequently, the address identifier of the permanent object loaded on the OBUUIEKUTO buffer is registered into address-mapping table 3a about the page which manages the permanent object concerned (steps S22 and S23). And said address identifier refers a permanent object (step S13).

[0059] As explanation was given [ above-mentioned ], it can set up based on the maintenance cost applied [ any shall be set as a permanent object between a location identifier or an object identifier, and ] in order to maintain referential integrity based on the attribute of a permanent object. Moreover, since the address with which the permanent object concerned is arranged, and the identifier set as the permanent object concerned are managed when the permanent object is loaded on the object buffer, the permanent object concerned arranged on the object buffer can be referred to, and while duplicate loading of newly reading from a secondary storage is avoidable, it can refer to quickly.

[0060] Moreover, the identifier group about self can be quickly obtained from the inside of an object table with the address currently held when the permanent object concerned refers to self behind since said permanent object held the address of the entry on the object table about the permanent object concerned when the permanent object to which the object identifier was set was loaded on an object buffer. Moreover, when making the backed up permanent object restore, the object identifier set as the permanent object at the time of backup can be set up as an object identifier set as a permanent object.

[0061] In addition, although the address-mapping table managed the location identifier and the address of a permanent object for every page, reduced the range searched for an address identifier based on a location identifier and was improving retrieval effectiveness in the above-mentioned example, you may make it manage about a permanent object with all location identifiers not only on the above but on one address-mapping table.

[0062]

[Effect of the Invention] While being able to fall a maintenance cost since either the location identifier or the object identifier was set as the permanent object based on the maintenance cost applied in the permanent object management method concerning this invention in order to maintain the referential integrity based on the attribute of a permanent object as explained above, reference effectiveness can also improve.

[0063] Moreover, since the address with which the permanent object concerned is arranged, and the identifier set as the permanent object concerned were managed in the permanent object management method concerning this invention when a permanent object was loaded on main storage, the permanent object concerned arranged on main storage can be referred to quickly, duplicate loading of newly reading from a secondary storage can be avoided, and main storage can be used effectively.

[0064] Moreover, in the permanent object management method concerning this invention, since it was made to make the address of the entry of the object table about the permanent object concerned hold to said permanent object when the permanent object to which the object identifier was set was loaded on main storage, the identifier group about self can be quickly obtained from the inside of an object table with the address currently held when the permanent object concerned refers to self behind, and processing effectiveness improves.

[0065] Moreover, in the permanent object management method concerning this invention, since it enabled it to set up the object identifier set as the permanent object at the time of

backup as an object identifier set as a permanent object, it is reproducible like the time of backing up the referential integrity between each permanent OBUJIEKU in the object oriented database after restoring the permanent object concerned.

[Translation done.]

\* NOTICES \*

JPO and NCIPI are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

## DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1] It is the block diagram of the object oriented database which enforces the permanent object management method concerning this invention.

[Drawing 2] It is a block diagram explaining the permanent object management method concerning this invention.

[Drawing 3] It is a flow chart explaining generation actuation of a permanent object.

[Drawing 4] It is drawing explaining the reference approach of a permanent object.

[Drawing 5] It is a flow chart explaining the procedure of referring to a permanent object.

[Drawing 6] It is drawing explaining the permanent object management method of the object oriented database which manages a permanent object only by the object identifier concerning the conventional example.

[Description of Notations]

1 ... Application program 2 [ 3o / 4 / 6 ... The page buffer Management Department, 7 ... Disk input/output section, / ... A page buffer 5 ... Permanent object Management Department / .. An object table, 3a .. Address-mapping table ] ... A secondary storage, 3 ... Object buffer

[Translation done.]

\* NOTICES \*

JPO and NCIPI are not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

## DRAWINGS

[Drawing 1]

[Drawing 4]

[Drawing 2]

[Drawing 3]

[Drawing 5]

[Drawing 6]

[Translation done.]

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**